REMARKS

Claim 1 is amended to clarify the present invention with regard to the cited art. The term "application program module" in Claim 1 is revised to --application program-- for consistent terminology throughout the claims. Claims 1-10, 13-20, 26, and 27 remain in the application, with no claim previously allowed.

Claims 1-6 and 26 are rejected as unpatentable over *Hetherington* (US 6,396,515) in view of newly-cited *Bryan* (US 6,697,089). The rejection acknowledges that *Hetherington* does not teach determining whether a boot of the application program is the first boot in order to set the language, but asserts that *Bryan* teaches "determining if it is the first boot of the application module in order to set the language [col. 4, lines 45-54]". The applicant respectfully traverses this rejection and the interpretation of *Bryan* on which the rejection is based.

Claim 1 is amended to point out that determining whether a boot is the *first boot* of the application program takes place in response to the boot of the application program. If that boot *is* determined to be the first boot of that application program, then a plurality of default language settings of the application program are set equal to the user interface language of the operating system. A subsequent boot of the application program enables the language settings previously set as the default language settings for that application.

Bryan teaches making user-selectable grammar and semantics available for an application program, especially for automated document delivery systems (column 4, lines 21-23). Accordingly to *Bryan*, user-selectable grammar and semantics are set during a "first time experience" procedure (column 4, lines 38-41) determining whether a run is the first time run of the application (column 4, lines 47-48). If so, the "human

user" is asked whether he wishes to customize the program interface. If he does not, the installation program continues until the installation is complete (column 4, lines 53-58).

Bryan emphasizes that customizing the user interface of an application program is dependent on the user's wishes. Not only at first installation, but also throughout use of the program, interactive dialog boxes are displayed to the user, who may choose various grammar and semantic routines presented by the program (column 5, lines 11-37). Alternatively, the user may defer customizing the dialog boxes during installation. Furthermore, the software vendor of that program may periodically query the installed application program to upload the user's customized the grammar and semantics, for analyses by the software vendor (column 6, lines 29-46).

What *Bryan* does not teach, however, is setting a plurality of default language settings of the application program to equal the user interface language of the operating system, in response to a first boot of the application program. That setting of default language settings thus occurs without querying the user or requiring other user interaction, in the method defined by Claim 1. That method thus is not only novel, but would have remained untaught to one of ordinary skill who knew of *Hetherington* and *Bryan* but not the contribution of the present inventor.

It should be recalled that a goal of the present invention is setting default language settings of the application program equal to a user interface language of the operating system, at first boot of the application program. Using the present invention, if the user runs any language version of a multi-language application program on a particular user-interface language of the operating system, that application behaves more like the localized version of the application for that operating system. For example, installing an English version of an application on a Japanese user-interface operating system will set

the installed language of the application to Japanese and will set various other defaults to behave like the Japanese version of the application (page 4, line 20 - page 5, line 3 of the specification). *Bryan* simply does not teach or suggest that invention.

Furthermore, while *Bryan* gives the "human user" opportunities to select grammar and semantics of an application, that opportunity necessarily presumes the user has at least some fluency in the application language presented on-screen. Considering as an example an English version of an application first booted with a Japanese operating system, or vice versa, *Bryan* would present a monolingual "human user" with semantic/grammar options in a language —and an alphabet— incomprehensible to that user. The present invention avoids or alleviates that comprehension barrier by setting default language settings of the application program to the user interface language of the operating system, in response to determining a first boot of the application program. One of ordinary skill, knowing only of *Hetherington* and *Bryan*, would remain unaware of the problem confronted by the present applicant and unaware of the applicant's solution to that problem as recited in Claim 1. Accordingly, that claim and the claims depending therefrom are patentable over the combination of *Hetherington* and *Bryan*.

Dependent Claim 26 states that the default language settings of the application program are set equal to the user interface language, in response to determining a first boot, only if registry key values for the language settings of the application program are currently at "Off" instead of "ExplicitOff". (That feature of the invention is discussed at page 10, lines 4-10 and page 11, lines 1-11.) The rejection of Claim 26 asserts that *Bryan* teaches that default language settings are set in response to any number of selection parameters such as a registry key, citing column 4, lines 36-58 of *Bryan*. However, a close reading of *Bryan* fails to find any mention of a registry key for any purpose at all,

let alone the specific limitations set forth in Claim 26. Furthermore, *Bryan* teaches only choosing grammar and semantics settings, but not a default language setting as asserted in the rejection of Claim 26, and presumably would have no reason to make any such user-selectable option depend on a particular registry key as claimed. That claim, combined with the limitations of parent Claim 1, defines a specific method whereby default language settings of an application program are set equal to the user interface language in response to a first boot, and if registry key values are at a certain setting. The elements of this claimed method simply are not found in *Bryan* and are not taught by that reference. Accordingly, Claim 26 is patentable thereover.

Claim 7-10, 13-20 and 27 are rejected as unpatentable over *Hetherington* in view of *Bryan*, further in view of *Kim* (US 6,014,616). The applicant traverses these rejections for the reasons set forth above with regard to *Hetherington* and *Bryan*. As to dependent Claim 27, requiring that adjusting the value of the registry keys associated with plural editing languages is performed only if the registry key values are at "Off" instead of "ExplicitOff", the applicant again points out that *Bryan* contains no mention whatsoever of registry key values, let alone the specific limitations concerning those values, and interacting with other steps of the method defined in Claim 27. Accordingly, that claim would not have been obvious over the applied art.

The foregoing is submitted as a complete response to the Office Action identified above. The Applicant respectfully submits that all claims remaining in this application are patentable over the art of record and solicits a notice of allowance to that effect.

Respectfully submitted,

MERCHANT & GOULD

Date: September 21, 2005

Roger T. Frost
Reg. No. 22,176

Merchant & Gould, LLC
P.O. Box 2903
Minneapolis, MN 55402-0903
Telephone: 404.954.5100